



Universidad Abierta Interamericana

Programación II (Optativa)

Código de materia 25

Prof. Titular: Ing. Gastón Matías Weingand

Facultad de tecnología informática

UNIVERSIDAD ABIERTA INTERAMERICANA



Universidad Abierta Interamericana

Guía de Revisión

Introducción a Python

1. ¿Qué diferencias existen entre un programa y un proyecto?
2. ¿Qué es el entorno integrado de desarrollo?
3. Enumere y explique las diferencias entre la programación estructurada, orientada a objetos y funcional.
4. ¿Qué es una abstracción? ¿Qué es un tipo?
5. Defina clases, objetos, herencia y polimorfismo
6. ¿Qué es python? ¿Por qué se dice que es semi interpretado?
7. ¿Por qué es importante conocer python? ¿Es adecuado para cualquier tipo de aplicaciones?
8. ¿Cómo es la gestión de tipos de python? ¿Fuerte o débil? ¿Estático o Dinámico?

Clases y Objetos

9. ¿Cuáles son las estructuras condicionales y de repetición que ofrece el lenguaje?
10. ¿Cómo se define una clase?
11. ¿Qué diferencia a una clase clásica con una de “nuevo estilo”?
12. ¿Cómo se instancia una clase para crear un objeto?
13. ¿Cómo se gestiona el encapsulamiento?
14. ¿Cómo y para qué se utilizan el constructor y el destructor?
15. ¿Para qué se utiliza el método `__init__(self, args)`?
16. ¿Cómo se implementa una herencia simple y múltiple?
17. ¿Qué métodos especiales son los más utilizados?
18. ¿Cómo se maneja el polimorfismo?

Programación funcional y excepciones

19. ¿Qué es una función de orden superior?
20. ¿Qué es una iteración de orden superior?
21. ¿Para qué se utilizan las funciones map, filter y reduce?
22. ¿Para que sirve el operador lambda?
23. ¿Qué son las list comprehensions?
24. ¿Qué son las expresiones generadoras?
25. ¿Qué es un decorador?
26. ¿Cómo se gestionan las excepciones?



Universidad Abierta Interamericana

Archivos y expresiones regulares

- 27. Diferencias entre módulo y paquete.
- 28. Entrada y salida estándar. Parámetros de línea de comandos. Ejemplos.
- 29. ¿Cómo se gestiona la escritura y lectura de archivos?
- 30. Movimiento del puntero de escritura. Ejemplo.
- 31. ¿Cómo se tratan los errores en la gestión de archivos?
- 32. Uso de expresiones regulares. ¿Qué es el módulo re? Ejemplos.
- 33. ¿Qué flags son los más utilizados?
- 34. ¿Qué es un RegexObject?

Librerías e introducción al machine learning

- 35. ¿Para qué sirven las librerías? Pygame, RE, Collections, NumPy, SQLAlchemy, Request y Pillow
- 36. ¿Qué es el machine learning?
- 37. ¿Qué es el deep learning?
- 38. Defina los conceptos de: Aprendizaje. Perceptrón. Función de activación. Red neuronal. Back propagation.
- 39. ¿Para que se utilizan los framework Anaconda y miniconda? ¿Diferencia entre ambos?



Universidad Abierta Interamericana

Guía Trabajos Prácticos

Introducción a Python

1. Realice un cuadro sinóptico donde se expliquen las ventajas y desventajas en PE, POO y Funcional.
2. Realice un cuadro sinóptico con las definiciones de clase, objeto, polimorfismo y herencia

Clases y Objetos

3. Escriba en un mismo código de ejemplo las sentencias condicionales y de iteración para cada estructura conocida del lenguaje: [if, elseif y else] [while y for].
4. Realice el código necesario para definir una clase
5. Realice el código necesario para instanciar dicha clase
6. Realice un ejemplo en código donde se demuestre el encapsulamiento
7. Realice un ejemplo en código para herencia simple y múltiple
8. Realice un ejemplo en código que permita polimorfismo
9. Realice un ejemplo en código donde se utilicen los métodos especiales init, new, del, str y cmp.

Programación funcional y excepciones

10. Realice un ejemplo en código para el uso de diccionarios, cadenas y listas.
11. Realice un ejemplo en código para el uso de una función de orden superior
12. Realice un ejemplo en código para cada función de: map, filter y reduce
13. Realice un ejemplo en código para el uso de una función lambda
14. Realice un ejemplo en código para el uso de una list comprehension
15. Realice un ejemplo en código para el uso de una expresión generadora
16. Realice un ejemplo en código para el uso de una función decoradora
17. Realice un ejemplo en código para el tratamiento de una excepción



Universidad Abierta Interamericana

Archivos y expresiones regulares

18. Genere un código que permita la creación y el uso de un módulo.
19. Agregue el uso de paquetes al código anterior.
20. Genere un código para la creación, apertura, grabación, lectura y cierre de un archivo
 - a. de texto
 - b. de datos binarios
21. Genere el código para tratar los errores de la gestión de archivos del punto anterior.
22. Genere un ejemplo donde aplique expresiones regulares en la lectura (Línea a línea) de un archivo X (Por ejemplo, encontrar direcciones de email u otro patrón que usted desee), guarde las líneas que contienen su patrón en un archivo y en otro las líneas que no lo contiene.

Librerías e introducción al machine learning

23. Genere códigos de ejemplo utilizando cada una de las bibliotecas: Pygame, RE, Collections, NumPy, SQLAlchemy, Request y Pillow.
24. Genere el código para que un perceptrón pueda aprender las compuertas AND, OR, NOT, XOR y XNOR.
25. Diseñe el código para que un vehículo arduino/micro:bit pueda aprender a evitar obstáculos



Universidad Abierta Interamericana

Guía de Abordaje

INTRODUCCIÓN A PYTHON

Raúl González Duque - “Python para todos”. 1ª Ed. España. Creative Commons 2015. Capítulos 1 y 2.
Eric Matthes - “Python Crash Course” 2ª Ed. USA. No Starch Press 2019. Introducción y Capítulo 1

CLASES Y OBJETOS

Raúl González Duque - “Python para todos”. 1ª Ed. España. Creative Commons 2015. Capítulos 7 y 8
David Beazley, Brian K. Jones - “Python Cookbook” 3ª Ed. USA. OReilly Media 2013. Capítulo 8.
Eric Matthes - “Python Crash Course” 2ª Ed. USA. No Starch Press 2019. Introducción y Capítulo 9

PROGRAMACIÓN FUNCIONAL Y EXCEPCIONES

Raúl González Duque - “Python para todos”. 1ª Ed. España. Creative Commons 2015. Capítulos 9 y 10
David Beazley, Brian K. Jones - “Python Cookbook” 3ª Ed. USA. OReilly Media 2013. Capítulo 10 y 14.
Eric Matthes - “Python Crash Course” 2ª Ed. USA. No Starch Press 2019. Introducción y Capítulo 8 y 10

ARCHIVOS Y EXPRESIONES REGULARES

Raúl González Duque - “Python para todos”. 1ª Ed. España. Creative Commons 2015. Capítulos 11, 12 y 13.
David Beazley, Brian K. Jones - “Python Cookbook” 3ª Ed. USA. OReilly Media 2013. Capítulo 2, 5 y 10.
Eric Matthes - “Python Crash Course” 2ª Ed. USA. No Starch Press 2019. Introducción y Capítulo 10

LIBRERÍAS E INTRODUCCIÓN AL MACHINE LEARNING

Aurélien Géron – “Hands-On Machine Learning with Scikit-Learn & TensorFlow”. 1ª Ed USA. OReilly Media 2017. Capítulo 10